

浮動小数点データの圧縮方法

“超並列計算時代の圧縮技術”

概要:

数値シミュレーションや画像処理といった様々な分野においてGPU等を利用した並列計算が当たり前になってきています。本研究では、超並列環境下で限られたメモリ領域を有効に活用したり、通信時のデータ伝送コストを削減したりするために、下記の特徴を持つ圧縮技術を開発しました。

- ・ 浮動小数点数や倍精度浮動小数点数で構成されるデータを効率よく圧縮
- ・ 並列処理することにより高速圧縮・高速解凍
- ・ データの内容（規則性や性質）にあわせて容易にカスタマイズ

【研究のねらい】

GPGPU（General-Purpose computing on Graphical Processing Unit）の普及に代表されるように、学協会や産業界において並列計算は一般的なものになりました。通常、並列計算の有効性は扱うデータの量が多いほど顕著に現れます。しかし、メモリ容量や通信帯域は限られているため、せっかくの演算能力を最大限に引き出すことができない場合があります。そこで本研究では、超並列環境において実行可能な浮動小数点データの圧縮方法を開発しました。

【研究内容と成果】

1. 図 1 に示すように、入力データ列中の値 d_n に対する予測値 p_n を $d_0 \sim d_{n-1}$ の値を使って導出します。このとき、データの内容に適した予測器を用いることで p_n の精度を高めることができます。例えば、正弦波的に値が変化することが多いデータの場合は多項式補外を用います。
2. d_n と p_n の排他的論理和を取ります。このとき、上位 bit に 0 が多く現れ、予測精度が高いほど下位 bit まで 0 が並びます。この 0 を、0 の続く数に置き換えることで容量を圧縮します。図 2 の例では、上位の 16bit を 4bit (0010 : 0 が 2byte 続くことを意味する) に圧縮しています。

基本原理は上記の通りですが、圧縮率を高めるため、データを並び替えるなど本研究ではいくつかの工夫をさらに施しています。表 1 に示すように、本手法は従来手法よりも高い圧縮率を実現しています。

【研究成果の活用】

- ・ 数値シミュレーションデータの保存やチェックポイントング
- ・ 大容量センシングデータの伝送効率向上
- ・ ビッグデータ解析
- ・ 超並列計算（GPGPU 等）を利用した企業との共同研究

表 1 圧縮率(圧縮後の容量/元の容量)の比較

| | Data1 | Data2 | Data3 | Data4 |
|----------|---------------|---------------|---------------|---------------|
| gzip | 0.9395 | 0.8607 | 0.9456 | 0.9572 |
| bzip2 | 0.9592 | 0.8524 | 0.9719 | 0.9836 |
| FPC(9) | 0.8806 | 0.8730 | 0.9522 | 0.9406 |
| pFPC(9) | 0.8838 | 0.8739 | 0.9561 | 0.9416 |
| GFC | 0.9165 | 0.9137 | 0.9869 | 0.9945 |
| proposed | 0.8411 | 0.8443 | 0.9197 | 0.8728 |

入力データ列 (M+1個の浮動小数点数)

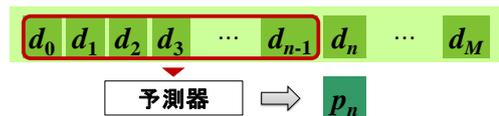


図 1 予測値の導出



図 2 圧縮の原理